

tradeoffs can be made intelligently in cases where CTQs compete with each other. These files will be described in more detail later.

[0053] The five files that the user creates are named according to a stereotyped naming pattern. The user specifies the root name for the files, minus any extension, and all the files get named using the root name plus specific filename extensions for each file. For instance, if the user picks the root name "c348", then the five files will be named "c348.fileWrapper", "c348.tplt", "c348\_master.exic", "c348.vdf", and "c348.pol". The root name may be the name of a particular probe being simulated.

[0054] Once the Transducer Design Advisor has been used to create or modify these five files, the ultrasound simulator can be used as a DOE analysis object.

[0055] The Transducer Design Advisor is similar to a Windows "Wizard". There are no drop-down menus or toolbar icons, just a set of buttons that the user uses to navigate through a series of windows. Each window asks the user to specify the values of a few related parameters. All of the Transducer Design Advisor windows have the same set of window navigation buttons.

[0056] Depending on whether the user is creating a new set of files from scratch or editing an existing set, and depending on the particular window the user is in at the moment, some of these buttons will be disabled (grayed out) and some will be enabled. The Back, Next and Help (?) buttons are almost always available. The Next button is used to proceed to the next window in the series when the user has finished entering values on the current window. The Back button is used to return to the window the user most recently visited, or the window before that, and so on. The Finish button (when enabled) is used to zoom to the end of the program and save the user's work if the user is editing an existing set of files and only needs to make a few changes.

[0057] If the user is editing an existing file set and the user wants to quickly move to a certain window to make changes, the GoTo... button can be used to select the window the user wants to jump to, instead of having to tediously click the user's way from window to window. This button is only enabled when the user is editing existing files, not when the user is creating a new set of files from scratch.

[0058] The Transducer Design Advisor preferably takes the form of a "wizard" which aids the setup of simulation and optimization details. Its purpose is to increase design productivity by eliminating the possibility of invalid parameter sets. It also encapsulates considerable knowledge of the domain of probe design. A similar wizard forms part of the DOE controller block 26 (see FIG. 1).

[0059] The Transducer Design Advisor in accordance with the preferred embodiment has about 50 different windows (three of which can be seen in FIGS. 5, 6 and 11). FIG. 7 shows an example of the flow from window to window of this wizard. The designer can move forwards and backwards through the choices until the design is ready for optimization. However, for a given transducer type, the user will only have to visit a subset of the windows. For instance, a convex probe will use a different subset of the windows than a linear probe, and a multi-row active matrix array (AMA) probe will use a different subset than a single-row probe.

[0060] The Transducer Design Advisor is implemented in Java 2. This graphical user interface is designed to be a "Wizard" similar to Windows Wizards: that is, a complicated problem is broken down into bite-sized pieces, and these pieces are presented in separate windows. The user is guided through the process one window at a time, using "Next" and "Back" buttons to navigate from one window to another. In the preferred embodiment of the Transducer Design Advisor, the "Wizard" concept was extended with several innovations: (1) In addition to "Next" and "Back" buttons, there is a "GoTo" button which the user can click to move instantly to a selected

window. This button is only available when a simulation fileset is being edited; when a fileset is being created for the first time, or when major modifications are being done to an existing fileset, the "GoTo" button is not available. (2) When graphical or tabular results are being displayed, separate pop-up dialog windows are used. These windows are not part of the normal "Wizard" flow of control; instead, they are short detours from the main path. (3) The navigation paths from window to window vary depending on the user's inputs in previous windows. A central navigation database is implemented in a WindowNavigator object to control this flow of control. A small window navigation language was designed so that this database is compact but easy to maintain. During program maintenance, it is very easy to change the flow of control should additional windows be added or deleted, or should the navigation logic be changed. Changing the set of successor windows to a particular window does not require the source code for any of the windows to be changed; only the navigation database need be modified. The same database is used to support the "GoTo" button.

[0061] The Java source files that make up the Transducer Design Advisor application can be grouped into several broad categories, including top-level objects (the core objects that make up TDA windows, the central window navigation logic, and state maintenance); acoustic stack simulator objects (objects to support the 1D transducer model); animation objects (objects which support mouse-driven graphical animation); graphics objects (objects supporting line graphs, bar graphs, contour graphs and so on); phantom objects (objects supporting creating and editing phantoms for imaging); and window objects (one for every TDA window).

[0062] Referring to FIG. 10, the Main class (containing the main() method) in the Transducer Design Advisor is TdaExecutive. The main() method bootstraps the singleton instance of TdaExecutive, which creates the first window. All regular windows (not counting pop-up dialogs, help windows, and graphical output dialogs) are subclassed from TdaFrame.